#Age

library(tidyverse)

library(ggplot2)

library(reshape2)

library(car)

library(rstatix)

set.seed(100)

data <- data.frame(x = rnorm(100, 2, 1), y = rnorm(100, 1, 1))

data2 <- melt(data)

data3 <- lapply(data, function(x) get\_summary\_stats(data.frame(x)))

data3

# $x

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -0.272 4.58 1.94 1.39 2.66 1.26 0.974 2.00 1.02 0.102 0.203

#

# $y

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -1.14 3.17 0.927 0.568 1.45 0.878 0.648 1.01 0.796 0.08 0.158

data3 <- rbind(data3[[1]], data3[[2]])

data3[1] <- c("x", "y")

## Shapiro-Wilk normality test

lapply(data, function(x) shapiro.test(x))

# $x

#

# Shapiro-Wilk normality test

#

# data: x

# W = 0.98836, p-value = 0.535

#

#

# $y

#

# Shapiro-Wilk normality test

#

# data: x

# W = 0.98532, p-value = 0.3348

## Levene's Test

leveneTest(value~variable, data = data2)

# Levene's Test for Homogeneity of Variance (center = median)

# Df F value Pr(>F)

# group 1 4.4476 0.03621 \*

# 198

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

t.test(value~variable, data = data2, var.equal = T)

# Two Sample t-test

#

# data: value by variable

# t = 7.6613, df = 198, p-value = 8.012e-13

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7364913 1.2470521

# sample estimates:

# mean in group x mean in group y

# 2.002913 1.011141

t.test(value~variable, data = data2, var.equal = F)

# Welch Two Sample t-test

#

# data: value by variable

# t = 7.6613, df = 186.92, p-value = 9.657e-13

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7363983 1.2471452

# sample estimates:

# mean in group x mean in group y

# 2.002913 1.011141

wilcox.test(value~variable, data = data2)

# Wilcoxon rank sum test with continuity correction

#

# data: value by variable

# W = 7844, p-value = 3.711e-12

# alternative hypothesis: true location shift is not equal to 0

summary(aov(value~variable, data = data2))

# Df Sum Sq Mean Sq F value Pr(>F)

# variable 1 49.18 49.18 58.7 8.01e-13 \*\*\*

# Residuals 198 165.90 0.84

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.2) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.2) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_boxplot(alpha = 0.2) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.1) +

geom\_boxplot(alpha = 0.1) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot() +

geom\_violin(data = data2, aes(x = variable, y = value, color = variable, fill = variable), alpha = 0.1) +

geom\_errorbar(data = data3, aes(x = variable, ymin=mean-sd, ymax=mean+sd), width = 0.2)

####################################################################################################################################

#OS

if (!requireNamespace("survminer", quietly = TRUE))

install.packages("survminer")

library(survival)

library(survminer)

# data <- lung

# colnames(data)[5] <- "variable"

fit <- survfit(Surv(time, status) ~ variable, data = data)

print(fit)

# Call: survfit(formula = Surv(time, status) ~ variable, data = data)

#

# n events median 0.95LCL 0.95UCL

# variable=1 138 112 270 212 310

# variable=2 90 53 426 348 550

survdiff(Surv(time, status) ~ variable, data = data)

# survdiff(formula = Surv(time, status) ~ variable, data = data)

#

# N Observed Expected (O-E)^2/E (O-E)^2/V

# variable=1 138 112 91.6 4.55 10.3

# variable=2 90 53 73.4 5.68 10.3

#

# Chisq= 10.3 on 1 degrees of freedom, p= 0.001

fit2 <- coxph(Surv(time, status) ~ variable, data = data)

summary(fit2)

# Call:

# coxph(formula = Surv(time, status) ~ variable, data = data)

#

# n= 228, number of events= 165

#

# coef exp(coef) se(coef) z Pr(>|z|)

# variable -0.5310 0.5880 0.1672 -3.176 0.00149 \*\*

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#

# exp(coef) exp(-coef) lower .95 upper .95

# variable 0.588 1.701 0.4237 0.816

#

# Concordance= 0.579 (se = 0.021 )

# Likelihood ratio test= 10.63 on 1 df, p=0.001

# Wald test = 10.09 on 1 df, p=0.001

# Score (logrank) test = 10.33 on 1 df, p=0.001

# plot

ggsurvplot(fit = fit, data = data, pval = T)

####################################################################################################################################

# DFS

if (!requireNamespace("survminer", quietly = TRUE))

install.packages("survminer")

library(survival)

library(survminer)

# data <- lung

# colnames(data)[5] <- "variable"

fit <- survfit(Surv(time, status) ~ variable, data = data)

print(fit)

# Call: survfit(formula = Surv(time, status) ~ variable, data = data)

#

# n events median 0.95LCL 0.95UCL

# variable=1 138 112 270 212 310

# variable=2 90 53 426 348 550

survdiff(Surv(time, status) ~ variable, data = data)

# survdiff(formula = Surv(time, status) ~ variable, data = data)

#

# N Observed Expected (O-E)^2/E (O-E)^2/V

# variable=1 138 112 91.6 4.55 10.3

# variable=2 90 53 73.4 5.68 10.3

#

# Chisq= 10.3 on 1 degrees of freedom, p= 0.001

fit2 <- coxph(Surv(time, status) ~ variable, data = data)

summary(fit2)

# Call:

# coxph(formula = Surv(time, status) ~ variable, data = data)

#

# n= 228, number of events= 165

#

# coef exp(coef) se(coef) z Pr(>|z|)

# variable -0.5310 0.5880 0.1672 -3.176 0.00149 \*\*

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#

# exp(coef) exp(-coef) lower .95 upper .95

# variable 0.588 1.701 0.4237 0.816

#

# Concordance= 0.579 (se = 0.021 )

# Likelihood ratio test= 10.63 on 1 df, p=0.001

# Wald test = 10.09 on 1 df, p=0.001

# Score (logrank) test = 10.33 on 1 df, p=0.001

# plot

ggsurvplot(fit = fit, data = data, pval = T)

####################################################################################################################################

# PFI

if (!requireNamespace("survminer", quietly = TRUE))

install.packages("survminer")

library(survival)

library(survminer)

# data <- lung

# colnames(data)[5] <- "variable"

fit <- survfit(Surv(time, status) ~ variable, data = data)

print(fit)

# Call: survfit(formula = Surv(time, status) ~ variable, data = data)

#

# n events median 0.95LCL 0.95UCL

# variable=1 138 112 270 212 310

# variable=2 90 53 426 348 550

survdiff(Surv(time, status) ~ variable, data = data)

# survdiff(formula = Surv(time, status) ~ variable, data = data)

#

# N Observed Expected (O-E)^2/E (O-E)^2/V

# variable=1 138 112 91.6 4.55 10.3

# variable=2 90 53 73.4 5.68 10.3

#

# Chisq= 10.3 on 1 degrees of freedom, p= 0.001

fit2 <- coxph(Surv(time, status) ~ variable, data = data)

summary(fit2)

# Call:

# coxph(formula = Surv(time, status) ~ variable, data = data)

#

# n= 228, number of events= 165

#

# coef exp(coef) se(coef) z Pr(>|z|)

# variable -0.5310 0.5880 0.1672 -3.176 0.00149 \*\*

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

#

# exp(coef) exp(-coef) lower .95 upper .95

# variable 0.588 1.701 0.4237 0.816

#

# Concordance= 0.579 (se = 0.021 )

# Likelihood ratio test= 10.63 on 1 df, p=0.001

# Wald test = 10.09 on 1 df, p=0.001

# Score (logrank) test = 10.33 on 1 df, p=0.001

# plot

ggsurvplot(fit = fit, data = data, pval = T)

####################################################################################################################################

# Forest plot

library(ggplot2)

library(patchwork)

dat\_ci

# Characteristics y.pos hr low.ci high.ci

# 1 T stage 9 NA NA NA

# 2 T1&T2 8 2.448 1.523 3.936

# 3 T3 7 2.862 1.496 5.475

# 4 N stage 6 NA NA NA

# 5 N0 5 2.570 1.621 4.074

# 6 N1 4 1.570 0.621 2.074

# 7 M stage 3 NA NA NA

# 8 M0 2 2.590 1.629 4.118

# 9 M1 1 1.548 1.123 2.536

p2 <- ggplot() +

geom\_errorbar(data = dat\_ci, aes(y = y.pos, xmin = low.ci, xmax = high.ci), width = 0.25) +

geom\_point(data = dat\_ci, aes(y = y.pos, x = hr), colour = "#0066DD") +

geom\_line(aes(x = c(1, 1), y = c(-Inf, 9.5)), linetype = 2, size = 0.5) +

geom\_line(aes(x = c(-Inf,Inf), y = c(9.5, 9.5))) +

geom\_line(aes(x = c(-Inf,Inf), y = c(10.5, 10.5))) +

scale\_y\_continuous(limits = c(0.5,10.5)) +

theme\_void() +

theme(axis.text.x = element\_text(), axis.line.x = element\_line(),

axis.ticks.length.x = unit(0.1, "cm"), axis.ticks.x = element\_line())

dat\_text

# label\_name y.pos

# 1 Characteristics 10

# 11 T stage 9

# 2 T1&T2 8

# 3 T3 7

# 4 N stage 6

# 5 N0 5

# 6 N1 4

# 7 M stage 3

# 8 M0 2

# 9 M1 1

p1 <- ggplot() +

geom\_text(data = dat\_text, aes(x = 1, y = y.pos, label = label\_name), size = 2.5) +

geom\_line(aes(x = c(-Inf,Inf), y = c(10.5, 10.5))) +

geom\_line(aes(x = c(-Inf,Inf), y = c(9.5, 9.5))) +

scale\_y\_continuous(limits = c(0.5,10.5)) +

theme\_void() +

theme(axis.line.x = element\_line())

p1 + p2

####################################################################################################################################

#infiltrating cells

# library(tidyverse)

library(GSVA)

library(clusterProfiler)

library(org.Hs.eg.db)

library(data.table)

library(rtracklayer)

### ssGSEA ######

## table S1 - https://doi.org/10.1016/j.immuni.2013.10.003

## pdf -> table -> read

immunity <- read.csv("~/immunity-cell-gene.csv", header = T)

# CellType AffymetrixID Symbol Gene.Symbol ENTREZ\_GENE\_ID

# 1 aDC 205569\_at LAMP3 LAMP3 27074

# 2 aDC 207533\_at CCL1 CCL1 6346

# 3 aDC 210029\_at INDO IDO1 3620

# 4 aDC 218400\_at OAS3 OAS3 4940

# 5 aDC 219424\_at EBI3 EBI3 10148

# 6 B cells 204836\_at GLDC GLDC 2731

idx <- !immunity$CellType %in% c("Blood vessels", "Normal mucosa", "SW480 cancer cells", "Lymph vessels")

immunity <- immunity[idx,]

immunity <- immunity %>%

split(., .$CellType) %>%

lapply(., function(x)(x$ENTREZ\_GENE\_ID))

immunity <- lapply(immunity, unique)

## Ensembl download

anno <- import('~/Homo\_sapiens.GRCh38.101.gtf')

anno <- as.data.frame(anno)

anno <- anno[!duplicated(anno$gene\_id),]

anno <- merge(anno, gene\_symbol, by = "gene\_name")

anno <- rbind(anno, data.frame(gene\_name = c("KIAA1324", "IGHA1"),

gene\_id = c("ENSG00000116299", "ENSG00000211895"),

ENTREZID = c("57535", "3492")))

anno <- anno[!duplicated(anno$gene\_id),] ### 37417

anno <- anno[, c("gene\_id", "ENTREZID")]

data <- fread("~/tpm.txt") %>%

rename("gene\_id" = "V1") %>%

left\_join(., anno, by = "gene\_id") %>%

filter(!is.na(ENTREZID)) %>%

select(-gene\_id) %>%

column\_to\_rownames("ENTREZID")

data <- log2(data + 1)

immu\_cell <- as.data.frame(gsva(as.matrix(data), immunity, method = "ssgsea"))

### cor plot

data <- read.table("~/file.txt", header = T)

# group aDC

# 1 1.13092315 0.4709550

# 3 0.55644003 0.1800251

# 4 0.44696904 0.3350859

# 5 0.05474605 0.1191767

# 7 0.61364297 0.1563856

# 8 0.41079217 0.4588979

colnames(data) <- c("x", "y")

ggplot(data, aes(x = x, y = y)) +

geom\_point() +

geom\_smooth(formula = y ~ x, method = "lm") +

theme\_bw()

####################################################################################################################################

# Paired sample differential gene expression

library(ggplot2)

library(reshape2)

set.seed(100)

data <- data.frame(x = rnorm(100, 2, 1), y = rnorm(100, 1, 1))

data$id <- 1:nrow(data)

data2 <- melt(data, id.vars = "id")

lapply(data, function(x) get\_summary\_stats(data.frame(x)))

# $x

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -0.272 4.58 1.94 1.39 2.66 1.26 0.974 2.00 1.02 0.102 0.203

#

# $y

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -1.14 3.17 0.927 0.568 1.45 0.878 0.648 1.01 0.796 0.08 0.158

data$diff <- data$x - data$y

shapiro.test(data$diff)

# Shapiro-Wilk normality test

#

# data: data$diff

# W = 0.98792, p-value = 0.5024

t.test(data$x, data$y, paired = T)

# Paired t-test

#

# data: data$x and data$y

# t = 7.2039, df = 99, p-value = 1.165e-10

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7186033 1.2649401

# sample estimates:

# mean of the differences

# 0.9917717

wilcox.test(data$x, data$y, paired = T)

# Wilcoxon signed rank test with continuity correction

#

# data: data$x and data$y

# V = 4287, p-value = 1.39e-09

# alternative hypothesis: true location shift is not equal to 0

ggplot(data = data2, aes(x = variable, y = value)) +

geom\_line(aes(group = id))+

geom\_point(aes(color = variable))

##################################################################################################################################### Pan-cancer unpaired differential gene expression

library(tidyverse)

library(ggplot2)

library(reshape2)

library(car)

library(rstatix)

set.seed(100)

data <- data.frame(x = rnorm(100, 2, 1), y = rnorm(100, 1, 1))

data2 <- melt(data)

data3 <- lapply(data, function(x) get\_summary\_stats(data.frame(x)))

data3

# $x

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -0.272 4.58 1.94 1.39 2.66 1.26 0.974 2.00 1.02 0.102 0.203

#

# $y

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -1.14 3.17 0.927 0.568 1.45 0.878 0.648 1.01 0.796 0.08 0.158

data3 <- rbind(data3[[1]], data3[[2]])

data3[1] <- c("x", "y")

## Shapiro-Wilk normality test

lapply(data, function(x) shapiro.test(x))

# $x

#

# Shapiro-Wilk normality test

#

# data: x

# W = 0.98836, p-value = 0.535

#

#

# $y

#

# Shapiro-Wilk normality test

#

# data: x

# W = 0.98532, p-value = 0.3348

## Levene's Test

leveneTest(value~variable, data = data2)

# Levene's Test for Homogeneity of Variance (center = median)

# Df F value Pr(>F)

# group 1 4.4476 0.03621 \*

# 198

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

t.test(value~variable, data = data2, var.equal = T)

# Two Sample t-test

#

# data: value by variable

# t = 7.6613, df = 198, p-value = 8.012e-13

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7364913 1.2470521

# sample estimates:

# mean in group x mean in group y

# 2.002913 1.011141

t.test(value~variable, data = data2, var.equal = F)

# Welch Two Sample t-test

#

# data: value by variable

# t = 7.6613, df = 186.92, p-value = 9.657e-13

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7363983 1.2471452

# sample estimates:

# mean in group x mean in group y

# 2.002913 1.011141

wilcox.test(value~variable, data = data2)

# Wilcoxon rank sum test with continuity correction

#

# data: value by variable

# W = 7844, p-value = 3.711e-12

# alternative hypothesis: true location shift is not equal to 0

summary(aov(value~variable, data = data2))

# Df Sum Sq Mean Sq F value Pr(>F)

# variable 1 49.18 49.18 58.7 8.01e-13 \*\*\*

# Residuals 198 165.90 0.84

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.2) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.2) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_boxplot(alpha = 0.2) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.1) +

geom\_boxplot(alpha = 0.1) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot() +

geom\_violin(data = data2, aes(x = variable, y = value, color = variable, fill = variable), alpha = 0.1) +

geom\_errorbar(data = data3, aes(x = variable, ymin=mean-sd, ymax=mean+sd), width = 0.2)

##################################################################################################################################### ROC—curve

library(tidyverse)

library(pROC)

library(ggplot2)

library(reshape2)

library(rstatix)

# data <- dat

data <- read.table("~/file.txt", header = T)

data$outcome <- factor(data$outcome, levels = c("group1", "group2"))

head(data)

# outcome a b c

# 1 group1 1.585855 1.1742805 2.674788

# 2 group1 2.205293 0.8619279 2.003079

# 3 group1 2.199554 2.3158722 1.281605

# 4 group1 1.241118 1.5746377 1.866428

# 5 group1 2.016992 1.9533336 1.847221

# 6 group1 2.391271 1.0891951 2.149648

data2 <- gather(data, key = "x", value = "value", -outcome)

data3 <- data2 %>%

group\_by(outcome, x) %>%

get\_summary\_stats(value)

data3

# outcome x variable n min max median q1 q3 iqr mad mean sd se ci

# 1 group1 a value 40 0.586 2.44 1.45 1.12 1.94 0.822 0.631 1.51 0.555 0.088 0.177

# 2 group1 b value 40 0.556 2.50 1.8 1.14 2.21 1.07 0.777 1.65 0.619 0.098 0.198

# 3 group1 c value 40 1.05 2.95 2.07 1.77 2.44 0.665 0.495 2.04 0.527 0.083 0.169

# 4 group2 a value 32 0.129 1.99 0.996 0.623 1.50 0.878 0.633 0.998 0.554 0.098 0.2

# 5 group2 b value 32 0.022 1.96 0.779 0.325 1.18 0.86 0.624 0.843 0.572 0.101 0.206

# 6 group2 c value 32 0.03 1.89 0.923 0.421 1.39 0.969 0.706 0.932 0.546 0.097 0.197

roc1 <- roc(response = data$outcome, predictor = data$a)

# Call:

# roc.default(response = data$outcome, predictor = data$a)

#

# Data: data$a in 40 controls (data$outcome group1) > 32 cases (data$outcome group2).

# Area under the curve: 0.7328

roc2 <- roc(response = data$outcome, predictor = data$b)

# Call:

# roc.default(response = data$outcome, predictor = data$b)

#

# Data: data$b in 40 controls (data$outcome group1) > 32 cases (data$outcome group2).

# Area under the curve: 0.8234

roc3 <- roc(response = data$outcome, predictor = data$c)

# Call:

# roc.default(response = data$outcome, predictor = data$c)

#

# Data: data$c in 40 controls (data$outcome group1) > 32 cases (data$outcome group2).

# Area under the curve: 0.9242

ci.auc(roc1)

# 95% CI: 0.6171-0.8485 (DeLong)

ci.auc(roc2)

# 95% CI: 0.7303-0.9165 (DeLong)

ci.auc(roc3)

# 95% CI: 0.8679-0.9805 (DeLong)

coords(roc1, x = "best", ret="all")

# threshold specificity sensitivity accuracy tn tp fn fp npv ppv fdr fpr tpr tnr fnr

# threshold 1.115538 0.75 0.625 0.6944444 30 20 12 10 0.7142857 0.6666667 0.3333333 0.25 0.625 0.75 0.375

# 1-specificity 1-sensitivity 1-accuracy 1-npv 1-ppv precision recall youden closest.topleft

# threshold 0.25 0.375 0.3055556 0.2857143 0.3333333 0.6666667 0.625 1.375 0.203125

roc.test(roc1, roc2, reuse.auc=FALSE, method = "delong")

# DeLong's test for two correlated ROC curves

#

# data: roc1 and roc2

# Z = -1.2406, p-value = 0.2148

# alternative hypothesis: true difference in AUC is not equal to 0

# sample estimates:

# AUC of roc1 AUC of roc2

# 0.7328125 0.8234375

plot.roc(roc1)

data1 <- data.frame(group = "a",

x = 1-roc1$specificities,

y = roc1$sensitivities)

data1 <- data1[order(data1$x, data1$y),]

tmp <- data.frame(group = "b",

x = 1-roc2$specificities,

y = roc2$sensitivities)

tmp <- tmp[order(tmp$x, tmp$y),]

data1 <- rbind(data1, tmp)

data1$group <- factor(data1$group, levels = c("a", "b"))

ggplot() +

geom\_line(data = data1, aes(x = x, y = y, colour = group)) +

labs(x = "1-Specificity (FPR)", y = "Sensitivity (TPR)")

####################################################################################################################################

#Stage

library(tidyverse)

library(ggplot2)

library(reshape2)

library(car)

library(rstatix)

set.seed(100)

data <- data.frame(x = rnorm(100, 2, 1), y = rnorm(100, 1, 1))

data2 <- melt(data)

data3 <- lapply(data, function(x) get\_summary\_stats(data.frame(x)))

data3

# $x

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -0.272 4.58 1.94 1.39 2.66 1.26 0.974 2.00 1.02 0.102 0.203

#

# $y

# # A tibble: 1 x 13

# variable n min max median q1 q3 iqr mad mean sd se ci

# <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>

# 1 x 100 -1.14 3.17 0.927 0.568 1.45 0.878 0.648 1.01 0.796 0.08 0.158

data3 <- rbind(data3[[1]], data3[[2]])

data3[1] <- c("x", "y")

## Shapiro-Wilk normality test

lapply(data, function(x) shapiro.test(x))

# $x

#

# Shapiro-Wilk normality test

#

# data: x

# W = 0.98836, p-value = 0.535

#

#

# $y

#

# Shapiro-Wilk normality test

#

# data: x

# W = 0.98532, p-value = 0.3348

## Levene's Test

leveneTest(value~variable, data = data2)

# Levene's Test for Homogeneity of Variance (center = median)

# Df F value Pr(>F)

# group 1 4.4476 0.03621 \*

# 198

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

t.test(value~variable, data = data2, var.equal = T)

# Two Sample t-test

#

# data: value by variable

# t = 7.6613, df = 198, p-value = 8.012e-13

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7364913 1.2470521

# sample estimates:

# mean in group x mean in group y

# 2.002913 1.011141

t.test(value~variable, data = data2, var.equal = F)

# Welch Two Sample t-test

#

# data: value by variable

# t = 7.6613, df = 186.92, p-value = 9.657e-13

# alternative hypothesis: true difference in means is not equal to 0

# 95 percent confidence interval:

# 0.7363983 1.2471452

# sample estimates:

# mean in group x mean in group y

# 2.002913 1.011141

wilcox.test(value~variable, data = data2)

# Wilcoxon rank sum test with continuity correction

#

# data: value by variable

# W = 7844, p-value = 3.711e-12

# alternative hypothesis: true location shift is not equal to 0

summary(aov(value~variable, data = data2))

# Df Sum Sq Mean Sq F value Pr(>F)

# variable 1 49.18 49.18 58.7 8.01e-13 \*\*\*

# Residuals 198 165.90 0.84

# ---

# Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.2) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.2) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_boxplot(alpha = 0.2) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot(data2, aes(x = variable, y = value, color = variable, fill = variable)) +

geom\_violin(alpha = 0.1) +

geom\_boxplot(alpha = 0.1) +

geom\_point(position = position\_jitter(0.3)) +

theme\_bw()

ggplot() +

geom\_violin(data = data2, aes(x = variable, y = value, color = variable, fill = variable), alpha = 0.1) +

geom\_errorbar(data = data3, aes(x = variable, ymin=mean-sd, ymax=mean+sd), width = 0.2)